APPENDIX A

% BEGINNING OF PSEUDO CODE

% compute scale factor A, and time constants a, b from physical system parameters

$$A = Vmax * Kt / (Re * Rm + Kt * Kb) * 1 * k;$$

$$p2 = 1/Jm/Ie * (-Ie * Rm - Re * Jm - sqrt(Ie^2 * Rm^2 - 2 * Re * Rm * Ie * Jm + Re^2 * Jm^2 - 4 * Kt * Kb * Ie * Jm)) / 2;$$

15
$$a = max(-p1,-p2)$$

 $b = min(-p1,-p2)$

% make initial guesses for step durations

% set maximum iteration count

Nmax = 1000;

25

for j = 1:Nmax

% save old values of step time intervals

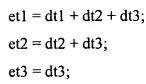
30 et3old = et3;

30

```
et2old = et2;
              etlold = etl;
              % iterate for switch times using fixed voltage level Vmax
 5
              et3 = -\log(1.0 / 2.0 - \exp(-\text{et1 * a}) / 2 + \exp(-\text{et2 * a})) / a;
              et2 = 1/b * log(2.0) + 3 * et3 - 1/b * log(2 * exp(1/A * b * X) * exp(et3)
                      * b) - sqrt( 4.0) * sqrt(\exp(1/A * b * X)) * \exp(et3 * b) *
                      sqrt(exp(1/A * b * X) + exp(et3 * b)^2 - 2 * exp(et3 * b));
              et1 = -(-2 * A * et2 + 2 * A * et3 - X) / A;
10
              if norm([et3old - et3 et2old - et2 et1old - et1], inf) <= eps * 2
                      break
              end
15
              if j==Nmax
                              error(['error - failure to converge after ', num2str(Nmax),'
                      iterations'])
              end
              end
20
              % round up pulse duration to nearest sample interval,
              % convert to intervals between steps to make sure that voltage
              % requirements will not increase (beyond Vmax).
              dt1=ceil((et1 - et2) / dt) * dt;
25
              dt2=ceil((et2 - et3) / dt) * dt;
              dt3=ceil((et3) / dt) * dt;
              et123 = [et1, et2, et3]
```

% convert back to total step duration.

<u>ļ</u>4



% In the following, the original constraints equations involving XF1, XF2, was and XF3 have been modified to include a variable voltage level applied at

% each step (instead of the fixed maximum (+/-) Vmax).

10 % The original equations for XF1, XF2, and XF3 follow:

%
$$XF_1(t_{end}) = V_0F_1(t_{tend} - t_0) - 2V_0F_1(t_{end} - t_1) + 2V_0F_1(t_{end} - t_2)$$

%
$$XF_2(t_{end}) = V_0F_2(t_{tend} - t_0) - 2V_0F_2(t_{end} - t_1) + 2V_0F_1(t_{end} - t_2)$$

%
$$XF_3(t_{end}) = V_0F_3(t_{tend} - t_0) - 2V_0F_2(t_{end} - t_1) + 2V_0F_1(t_{end} - t_2)$$

15 % And the modified equation including adjustable relative levels of voltage

% L1, L2 and L3 are:

%
$$XF_1(t_{end}) = L_1V_0F_1(t_{tend} - t_0) - L_2V_0F_1(t_{end} - t_1) + L_3V_0F_1(t_{end} - t_2)$$

%
$$XF_2(t_{end}) = L_1V_0F_2(t_{tend} - t_0) - L_2V_0F_2(t_{end} - t_1) + L_3V_0F_1(t_{end} - t_2)$$

% And the corresponding constraint equations are:

$$\%$$
 XF₁(t_{end}) = Finalpos

$$% XF_{2}(t_{end}) = 0$$

25 %
$$XF_3(t_{end}) = 0$$

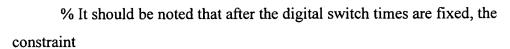
- % Where all of the times indicated have discrete values, e.g. corresponding to
 - % the controller update rate.

30

10

20

25



% equations derived from the equations above form a linear set of equations in

5 % the unknown relative voltage levels L1, L2 and L3 and any standard linear

% method can be used to solve for the relative voltage levels. In the equations

% for (L1, L2 and L3) that follow, the solution was obtained by algebraic % means (and are not particularly compact.)

% compute new relative voltage step levels

% L1, L2 and L3 are nominally assigned to "1", "-2" and "+2", respectively

L1 = s1 * s2;

20

exp(-et1 * a) - exp(-et2 * b) * et3 + exp(-et3 * b) * et1 * exp(-et2 * a) + exp(-et3 * a) * et1 + exp(-et2 * b) * et1 - exp(-et2 * b) * et1 * exp(-et3 * a) - et3 * exp(-et1 * b) * exp(-et2 * a) - exp(-et2 * a) * et1-exp(-et3 * b) * et1 - exp(-et3 * b) * et2 * exp(-et1 * a) + et3 * exp(-et1 * b) + et2 * exp(-et1 * a) + exp(-et3 * b) * et2 + et3 * exp(-et2 * a)) / A;

L2 = s1 * s2;

% convert accumulated voltage steps to sequential voltage level

V1 = Vmax * (L1);

25 V2 = Vmax * (L1 + L2);

V3 = Vmax * (L1 + L2 + L3);

% END OF PSEUDO CODE